1 Notation

We consider a certain key file, such as `2020-10-10.zip`.

- Enumerate the days a diagnosis key in the key file can be valid by $d \in \mathcal{D} := \{1, \dots, 13\}$, with $d = 1$ corresponding to the most recent date and $d = 13$ to the oldest date.
- Let $\mathcal{L} := \{1, \dots, 8\}$ be the set of possible risk levels.
- For day $d \in \mathcal{D}$ and level $l \in \mathcal{L}$ let n_{dl} be the number of keys valid on day d with level l.
- Let $\mathcal{T}$ be the set of all possible transmission risk matrices. To ease notation we use a transmission risk matrices instead of a transmission risk vector. A transmission risk matrix is a matrix indexed by $(d, l) \in \mathcal{D} \times \mathcal{L}$ with $t_{dl} = \begin{cases} 1, & \text{if risk level } l \text{ on day } d \text{ assumed,} \\ 0, & \text{else.} \end{cases}$

  In other words, the transmission risk matrix is the matrix of the indicator function of the transmission risk vector. An example of this correspondence between risk vector and risk matrix:

$$\begin{pmatrix} 6 \\ 8 \\ 8 \\ 8 \\ 5 \\ 3 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \leftrightarrow \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

- For $T \in \mathcal{T}$ let m_T be the number of users who uploaded their exposure and where the app assigns transmission risk matrix T.
- For a matrix X indexed by $\mathcal{D} \times \mathcal{L}$ we let $\operatorname{vec} X$ be the vector obtained by flattening the matrix by concatenating rows.
- Let $\boldsymbol{n} := \operatorname{vec} N$, where $N := (n_{dl})_{d \in \mathcal{D}, l \in \mathcal{L}}$. Furthermore $A := (\operatorname{vec} T^{(1)}, \dots, \operatorname{vec} T^{(r)})$ and $\boldsymbol{m} := (m_1, \dots, m_r)^T$ if $\mathcal{T}$ is enumerated as $\mathcal{T} = \{T^{(1)}, \dots, T^{(r)}\}$.

2 Problem

From observing $(n_{dl})_{d \in \mathcal{D}, l \in \mathcal{L}}$ we want to estimate $(m_T)_{T \in \mathcal{T}}$.

3 Model

If m_t is known for all $t \in T$, we can compute n_{dl} as follows:

$$n_{dl} = \sum_{T \in \mathcal{T}} m_T T_{dl},$$

or in linear operator form

$$\boldsymbol{n} = A\boldsymbol{m}.$$

4 Estimation problem

A natural way to estimate $\boldsymbol{m}$ from given $\boldsymbol{n}$ is by regression, which would amount to solving

$$\min_{\boldsymbol{m} \in \mathbb{Z}^r, \boldsymbol{m} \geq 0} \|\boldsymbol{n} - A\boldsymbol{m}\|_p^p,$$

with the typical choices $p = 2$ for ordinary least squares and $p = 1$ for least absolute deviation regression.

Both problem can be understand as closest vector problem (CVP) on lattices and can be solved by algorithms for CVP as well as by integer programming techniques. For $p = 2$ the problem is an integer quadratic problem with positive semidefinite hessian and $p = 1$ can be reformulated as linear programming (by introducing slacks for the absolute values).